D.R.Y.H. Low Power SRAM Chip

Daniel Klopp (dsk4x)
Roy Rabindarath (rpr6f)
Yao Yao (yy5v)
Hoa Nguyen (htn4x)

ECE 4332 – Fall 2009
University of Virginia

ABSTRACT

This paper provides a high level overview of the D.R.Y.H. 0.6 µm Low Power SRAM Chip presented to PICo as a solution to one of their needs. The chip is characterized by low power operation, with end-user variable clock and voltage settings for optimal run time performance.

1. INTRODUCTION

This paper depicts major innovations, design decisions and functional operation of D.R.Y.H.'s SRAM proving the effectiveness of our "proof of concept" IC, as explicitly stated in the proposal. Figures of less importance are omitted. All chip components are shown in Figure 1 and 2 below.

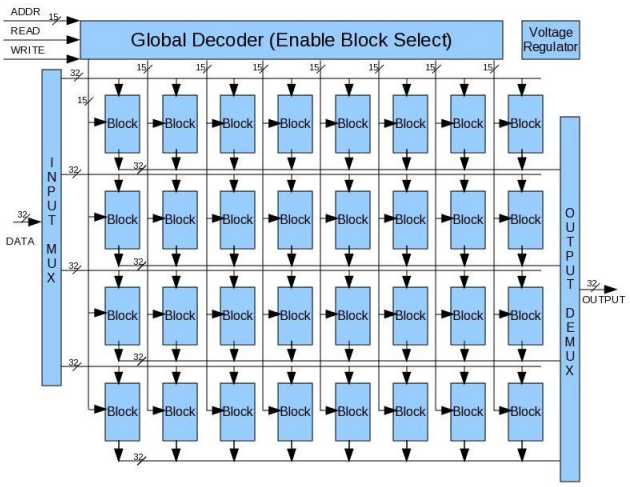

Figure 1: High Level Overview of SRAM

2. COMPONENTS

2.1 Miscellaneous Items

The Word Selection Multiplexer (see Fig. 2), Inpux Mux (see Fig. 1), Output Demux (see Fig. 1) are 3 micron passgates with basic control logic to feed the correct lines through.

The Write Drivers (see Fig. 2) were obtained from [2] (not shown) with BL and BLB from the Bitcells connected to two NMOS attached to ground. During a write, one NMOS would activate and force the BL or BLB to go low, while the other maintained its high from precharge, forcing a write operation. Each Write Driver is two Bitcells wide, so two rows of Write Drivers are stacked across each block, and metal is routed to the correct Bitcells below. Early on we merged the Write Driver and Sense Amps on the bottom of the block and shared the global data lines for reads and writes, but this did not work as the passgates needed upsizing and too much power was wasted passing a 1 and 0 through the passgates onto BL and BLB. So we sacrificed area in terms of separate global data read and write lines for 2-3 mW power savings at 2.5 Volts.

The chosen Row Decoder was an NOR-NAND gate implementation as shown in the textbook (pg 674, Fig 12-41). We chose this design because it reduced the area by 50% in comparison to an AND gate implementation. After a switch, static current was minimal, and this further reduced power. A dynamic decoder would have been faster, but given our clock speed and power constraints, this was irrelevant.

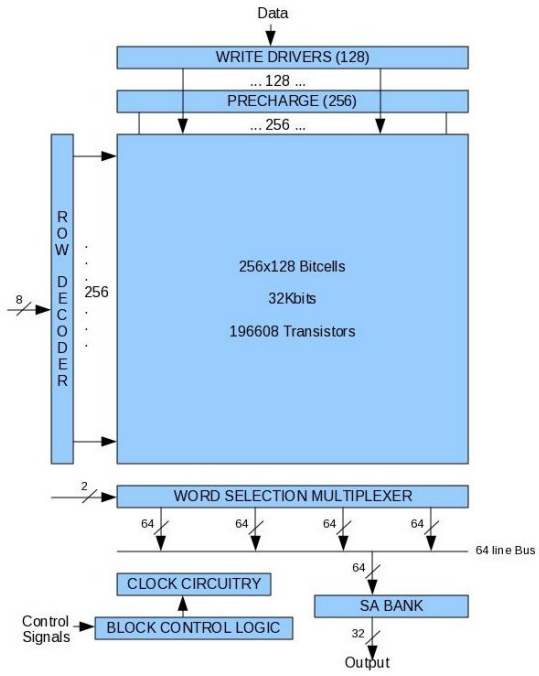

Figure 2: High Level Overview of Block

2.2 Bitcells

A standard 6 transistor cell was used with a CR of 1.7, PR of 1.0, and effective area of 140 um^2. The layout was chosen not only for

size reduction, but also for LVS compatibility with our schematics.

Figure 5 shows a plot of the bitcell's noise margins at 1.5, 2.0 and 2.5 volts. These data points were obtained from the circuit described in [1] (not shown).

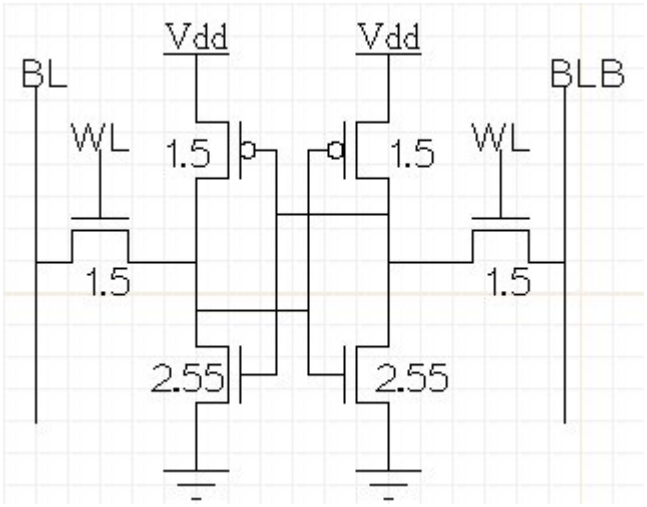

Figure 3: Bitcell Schematic

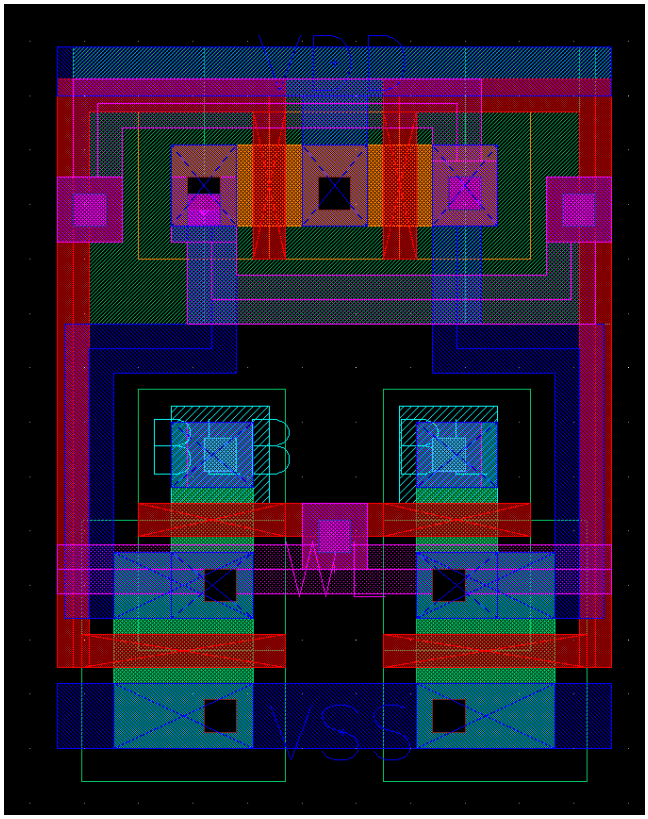

Figure 4: Bitcell Layout

2.3 Clock

D.R.Y.H.'s SRAM operates on both the rising and falling edge of the clock. As the Blocks precharge after the falling edge, the Global Decoder activates and feeds input data and address bits to the proper blocks. At the rising edge the Global Decoder turns off, and the Block performs its operations within 30 ns and waits until the next rising edge. These operations were timed for 1.5 Volt operation and can be scaled up to higher voltages if greater noise margins/reliability are concerned. Additionally, higher voltages can be obtained without wasting power by decreasing the clock period, as the block operates for a brief period of 30 ns and then "waits". Since power is the average energy over time, slower clocks will return massive power savings.

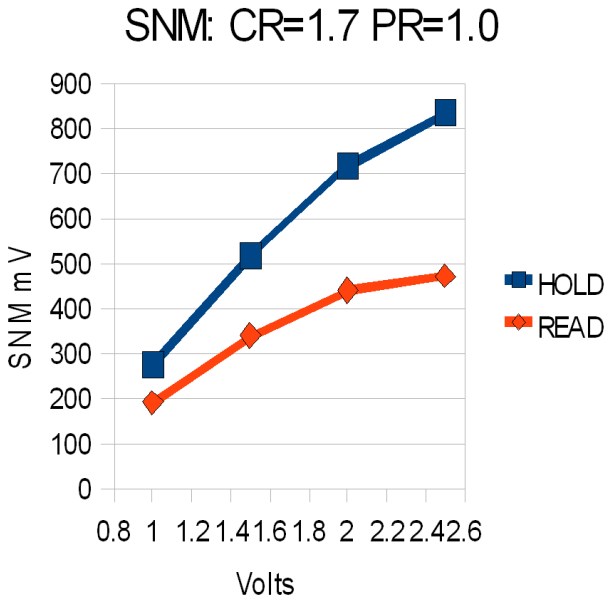

Figure 5 Worse Case Noise Margins (FS)

2.4 Control Logic & Clock Pulser

The control logic is not shown here, but it is a custom sized collection of logic gates that generate a NOP under all cases except when it receives the correct address bits and the XOR of read and write is 1. A NOP is synonymous with "low power" mode. For more details see the appendix.

The Clock Pulser is very complicated. Shown in Figure 6 is a high level overview of our functional circuit. There are two CustomDelay blocks, and they are not the same, each one is two inverters carefully sized to produce the necessary delay shown in the timing diagram (please see appendix). We opted for sizing the lengths of most of these components instead of adding more inverters for the same delay. This was done as it decreased power while increasing delay. The pulse generators are custom AND gates with customized inverters. Even the output buffers were custom sized to generate a high enough switching speed and current to drive their respective loads. Such sizing did not always optimize power reduction directly, though it did succeed indirectly by minimizing the greater lost charge in reads and writes.

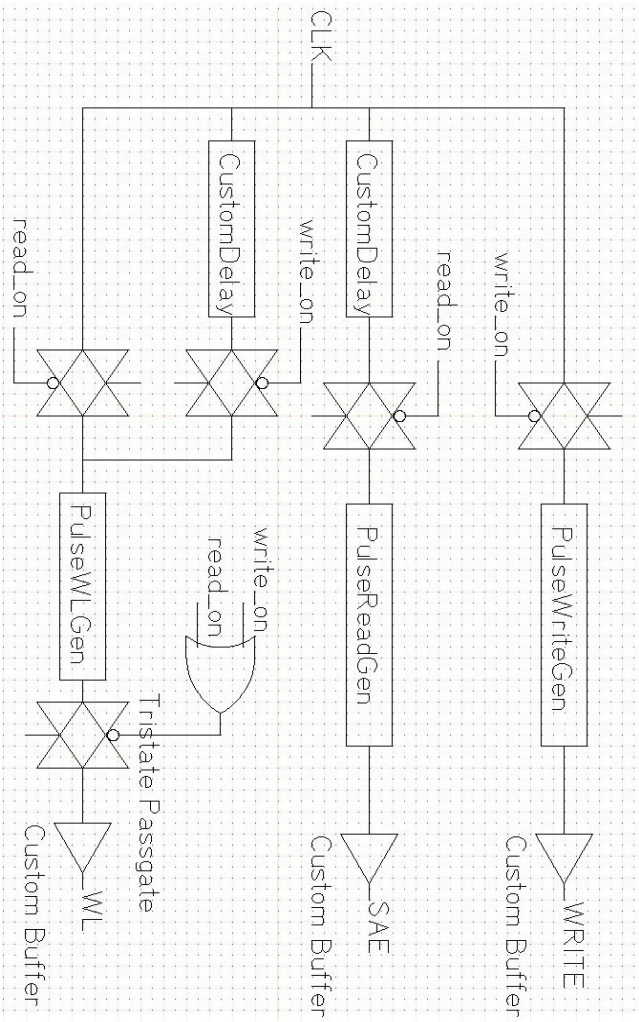

Figure 6: Clock Pulser (high level diagram)

2.5 Sense Amplifier

The Sense Amplifier design went through several versions. The first design was a simple driver referenced in [3], with undesirable static current. The second revision was provided by Professor Calhoun. Leftover charge in the cross coupled inverter cell made accurate sensing difficult. We wound up inserting two PMOS devices as shown in Figure 8.

Initial switching speed was phenomenal, acting well below a nanosecond. However, such operation rapidly exceeded 200 µA. Such high speed operation was not needed given our 60 ns clock speed, hence the switching speed was slowed (for detailed graphs of Sense Amp characterization, please see Appendix). Switching speed was 2 ns, and a 0.2 volt difference was necessary for a switch. This produced 60 µA of current per switching. Though the current draw was still high, it was deemed better this way, as the circuit is capable of running on a very slow clock. Hence, a single switching event would draw lots of current, but during steady state operation, effectively no power was drawn.

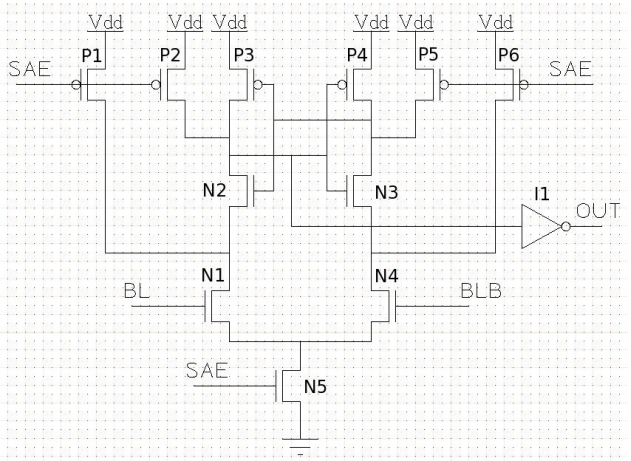

Figure 7: Sense Amplifier Schematic

2.6 Global Decoder

The Global Decoder was expensive in terms of power. Normal operation at 2.5 volts resulted in power losses exceeding 1 mW. Power was saved by running the device during the precharge cycle (falling edge of the clock) and only charging the global lines that needed charging. For example, if the block in the lower-right column is active, the other seven columns do not need to be charged except for one control signal (the enable signal, otherwise the inactive blocks may turn on if enable floats too high). The Global Decoder also feeds the proper two address bits to the Input Mux, ensuring only one row of data input lines are active, the rest are left to float.

3. Functionality / Issues

3.1 SKILL, Large Simulations, Models

All simulations after the semester midpoint were performed with Models of the actual circuit. Over 1700 lines of code were written to build the blocks and major components of the circuit (except layout, see section 3.2). A number of the built components cannot be simulated with existing UVA hardware. A several cycle simulation of the actual 256x128 bitcell block with all peripheral devices took 5 hours. When all power and currents were saved to compare the models with the actual block, Cadence ran out of memory and crashed.

Table 1: Functional Metrics of 1.5 Volt Circuit Model

Bitcell / Total Area	137.7 um^2 / 209 mm^2
Read / Write Power	658 µW / 795µW
Total Power	730 µW
All Delays	60 ns
Project Metric	6682

3.2 Layout

Because the bitcell block was small, peripheral units were made to exactly fit to the height and width. The write driver and column decoder were made to fit the narrow width of the block. Similarly, the row decoder used stacked Nand gates to fit the height of the block. All individual components successfully passed LVS check.

3.3 Commentary

As far as a "proof of concept" IC, this device is complete. We have working models at 1.5 and 2.5 volts, and the 1.5 volt model has been tested under 1.5, 1.6 and 1.7 volts. Device sizing enables circuit operation from at least 1.5-2.5 volts provided the Block Clock is adjusted (the pulses are configured for a small voltage range). Noise margins are shown in Figure 5, and PICo may choose the operating region they prefer given reliability constraints.

4. Features

4.1 Low Power Mode

The 4.5 - 5.5 voltage provided to chip is immediately fed into the Super Block's Voltage Regulator and downscaled to 1.5 volts for global consumption. Originally, we operated in two modes, 2.5 volts for the active block (read or write) and 1.5 for the inactive block. This was easier said than done as switching voltage states too early could incur unpredictable circuit behavior. Additionally a lot of extra area was wasted. However, it was noted that 0.5 um technology has negligible leakage current with respect to a read or write current loss. Careful measurements showed that the greatest power loss by many orders of magnitude came from the global decoder and an active block, hence the greatest power savings were obtained from operating the entire circuit at 1.5 volts. This doubled our clock period but reduced power consumption from 12 mW to 736 μW for the same operational sequence. We also saved area by not generating an additional voltage regulator for each block.

4.2 Pulsed Clocking

All blocks make use of a complicated pulse clocking mechanism (see Fig 6). Operational costs of the clock and decoder were not cheap, but the savings incurred from barely opening the word lines were vast. The clock works with the Block Control Logic in the next special feature: banking.

4.3 Banking

With 32 blocks and only one active at any time, there are many components in the 31 other blocks that are not necessary. Therefore, we opted to turn off those components. The Row Decoder, the Clock Pulser, Sense Amps, Writer Drivers and muxes were all deactivated when a block executes a NOP. The SRAM chip itself participates by deactivating the Global Decoder on the rising edge of the clock and only charging necessary data input lines and column control signals.

4.4 Byte Addressability

This feature was proposed in initial design reviews but eventually scrapped as its gain with respect to low power operation was not clear. The extra area and costs in power for the operating circuitry was not negligible. Additional delay would have been incurred from further control logic. In other words, though the feature was interesting on paper, it detracted from every metric used to gauge the product.

5. ACKNOWLEDGMENTS

Thanks goes especially to Professor Benton Calhoun in his early design help for the project.

6. REFERENCES

[1] Cabe, Adam. *Tutorial for finding Static Noise Margin using simulation.* https://venividiwiki.ee.virginia.edu/twiki/bin/view/Main/ToolsSimulationMemoryStaticNoiseMargin

[2] Pavlov, Andrei; Sachdev, Manoj. *CMOS SRAM Circuit Designa nd Parametric Test in Nano-Scaled Technologies*. Springer, US 2008.

[3] Mehta, Khanjan. *CSE 477: VLSI Project Home Page*. http://www.cedcc.psu.edu/khanjan/vssa.htm